

ISC 5228

Markov Chain Monte Carlo

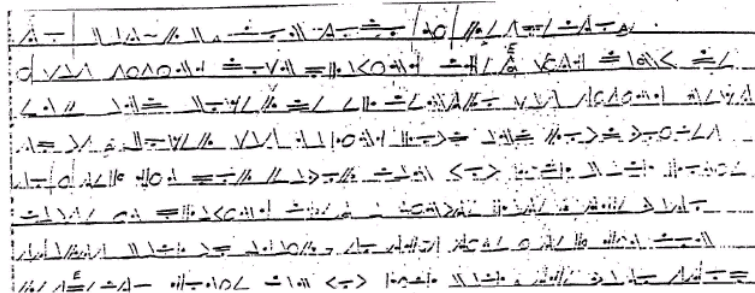
Fall 2015

Cryptography and Substitution Ciphers

1 Introduction

Perci Diaconis¹ narrates a fascinating incident:

Stanford's Statistics Department has a drop-in consulting service. One day, a psychologist from the state prison system showed up with a collection of coded messages.



Assuming a simple substitution cipher, where each letter (symbol) is mapped to a unique letter of the alphabet (example: $A \rightarrow S$, $B \rightarrow Z$, etc.), and Monte Carlo simulation, they were able to translate the message into:

to bat-rb. con todo mi respeto. i was sitting down playing chess with
danny de emf and boxer de el centro was sitting next to us. boxer was
making loud and loud voices so i tell him por favor can you kick back
homie cause im playing chess a minute later the vato starts back up again
so this time i tell him con respecto homie can you kick back. the vato
stop for a minute and he starts up again so i tell him check this out shut
the f**k up cause im tired of your voice and if you got a problem with it
we can go to celda and handle it. i really felt disrespected thats why i
told him. anyways after i tell him that the next thing I know that vato
slashes me and leaves. dy the time i figure im hit i try to get away but
the c.o. is walking in my direction and he gets me right dy a celda. so i
go to the hole. when im in the hole my home boys hit doxer so now "b" is
also in the hole. while im in the hole im getting schoold wrong and

We will try to mimic this kind of exercise in the project. For simplicity we will consider a unique 1-1 "substitution" mapping among the 26 letters of the alphabet. For example:

$$f : \{ 'A', 'B', \dots, 'Z', ' ' \} \rightarrow \{ 'S', 'Z', \dots, 'R', ' ' \}$$

We will ignore all punctuation, numbers, etc., except "space", which we will assume always maps into itself.

¹"Markov Chain Monte Carlo revolution", *Bull. Amer. Math. Soc.* 46 (2009), 179-205

1.1 Goal

The goal of this lab is to decode the “secret” phrase:²

BIU V DICT ZMEF VBZFU MZ NVJ OJGFUFL MCZI ZGMJ NIUDL IB JIUUIN VCL ZUIOQDF QK
ZGF HVUMJG JOUTFIC MZ UFEVMCFL V EVZZFU IB PICJMLFUVQDF LIOQZ NGFZGFU ZGF
PGMDL NIODL JOURMRF ZI QFVU VCK CVEF VZ VDD MC NGMPG PVJF MZ MJ JIEFNGVZ
EIUF ZGVC HUIQVQDF ZGVZ ZGFJF EFEIMUJ NIODL CFRFU GVRF VHHFVUFL IU MB ZGFK
GVL ZGVZ QFMCT PIEHUMJFL NMZGMC V PIOHDF IB HVTFJ ZGFK NIODL GVRF HIJFJJFL
ZGF MCFJZMEVQDF EFUMZ IB QFMCT ZGF EIJZ PICPMJF VCL BVMZGBOD JHFPMEFC IB
QMITUVHGK FYZVCZ MC ZGF DMZFUVZOUF IB VCK VTF IU PIOCZUK

In other words we seek the mapping f that helps us make sense of this phrase.

2 Approach

Before we consider the approach, let us try to fathom why an exhaustive brute force search is a bad idea. There are 26 letters. It can be shown that the number of different maps possible is $26! > 10^{26}$, which would take too long to explore.

One idea is to consider a significant corpus of the literature, and record *first order transitions* from a particular letter “x” to the letter “y”. Thus, we can try to estimate for example, how frequently the letter “U” follows letter “Q” etc. For this lab, I downloaded “War and Peace”, “Oliver Twist”, and “King James Bible” from Project Gutenberg.³ This yields a 27×27 table or matrix which is available as the file pairFreq.dat. In this matrix M_{ij} represents the transition frequency from the i^{th} letter of the alphabet to the j^{th} . Thus M_{13} represents the frequency with which the letter “A” is followed by the letter “C” in this corpus. For clarity, I will sometimes refer to M_{13} as $M(A, C)$.

We start with a randomly guessed mapping f . If N is the number of letters in the message, then we compute the *plausibility* or likelihood function,

$$\text{Pl}(f) = \prod_{i=1}^{N-1} M(f(s_i), f(s_{i+1})), \quad (1)$$

where s_i is the i^{th} letter of the encrypted message. One can think of this as the probability distribution of the different mappings f . To protect from underflow and overflow, we will work with the logarithm of $\text{Pl}(f)$.

3 Strategy

We can use Metropolis-Hastings to sample the probability distribution

1. Guess a mapping f

²There is a space after each word that ends a line in the text-block below.

³<https://www.gutenberg.org/>

2. Compute $\log \text{PI}(f)$ from eqn. 1.
3. Propose a new mapping f' by swapping two random letters.
4. Compute $\log \text{PI}(f')$.
5. Decide whether to accept or reject the mapping by comparing $\text{PI}(f')/\text{PI}(f)$
6. Repeat steps 3-6 until the phrase makes sense, or there is no subsequent change in the “deciphered” text.
7. If the message is still garbled, try a different initial guess f .

You will need to write functions to do the following tasks:

- Generate a random mapping f
- Compute $\log \text{PI}(f)$ given the message, f , and matrix \mathbf{M}
- Swap two letters to generate a proposed mapping f' from f .
- Compute Hastings ratio, and accept or reject f'

4 Deliverable

You are encouraged to work in teams of two. Write a detailed, and well-organized report (PDF format) with descriptions, code, figures, and sample runs etc. 20% of the grade for the project will be assigned based on the quality of the report.

Your MCMC simulation does not have to precisely decrypt the message. If it gets close enough, so that you can use your human skills to “read” the message, you should be good!

The report is due by 11:59PM on December 4, 2015.